**Performance Engineering**

Achieving good performance in coupled simulations is a daunting and a labor-intensive task. In this project, the University of Oregon team will deploy the TAU Performance System for performance engineering, verification, and validation tasks. TAU provides a comprehensive profiling and tracing toolkit for performance evaluation and tuning of HPC codes. It supports the source instrumentation of memory, I/O, and communication libraries. To fully observe the performance artifacts affecting the coupled MHD simulation running on leadership class DOE heterogeneous HPC platforms, it is necessary to measure the performance of individual components and the framework connecting the XGC1, NIMROD, and M3D-C1 codes. Parts of these applications will execute on accelerators that share memory with the compute nodes. Instrumenting applications that use NVIDIA and AMD GPUs and Intel Xeon Phi co-processor (aka MIC) accelerator nodes requires a common performance model in a tool that supports this diversity of accelerator devices in its instrumentation and measurement layers. As the coupled MHD codes are instrumented and executed on these hybrid nodes, data exchange between a host CPU and an accelerator connected by a slower PCI-X bus, and between compute nodes themselves will require careful orchestration and tracking of the asynchronous execution of kernels. Controlling and measuring the volume of data exchanged between these MHD components may be as relevant as observing the balance of computation, communication, and co-scheduling of components to reduce the overall wait states in the coupled simulation. When the instrumented kinetic and MHD components are executed concurrently in different spatial grids and computational nodes, common performance instrumentation events from multiple layers will flow into a common performance data repository in TAU. This data will be exported into profile files for subsequent analysis as well as stored as performance provenance data records in the output files collected at the end of the execution. The ADIOS layer will be instrumental in exposing the application data and the performance data for each execution. By providing the TAU performance provenance information as a component of the ADIOS output, it will help us classify and explain the variability in I/O performance due to system noise and use of shared I/O resources. TAU will interface with ADIOS to extend the current scope of the output data to retain this critical information. Profiling the I/O resources of the simulation will also help shed light on data exchange that will be optimized as we transition from a file system based approach to an in-memory approach that uses MPI and other technologies such as Kepler and Swift in the EFFIS 2.0 framework. The data transfer rate between codes residing on different spatial grids and computational nodes will be tracked using TAU to help optimize the parallel mesh-to-mesh coupling and interpolation. The performance data will be stored in a TAUdb database and used for scaling studies using TAU’s PerfExplorer cross-experiment analysis tool and the ParaProf 3D profile browser. Historical performance data will help us evaluate the efficacy of changes in the EFFIS 2.0 framework during the developmental stage. Specifically, we will evaluate and store performance data collected from the coupled routines, the framework used for coupling, and the I/O layer. The data will comprise of profiles that show the exclusive time spent at the routine, loop, and statement granularity and will track the volume of data. Runtime analysis of data will help identify peaks in the bandwidth of mesh exchanges and the I/O data volume by triggering context events that couple the calling context in the form of a program callstack with the value of the data that exceeds previously seen thresholds. These peaks will reveal fluctuations in performance and the code regions that are responsible for it. When key algorithmic changes are made to the coupling of these codes such as using the same mesh geometry between two components of the simulation, TAU will help quantify the effect of the change by comparing the baseline performance data with the updated components. Over the course of the project, we will collect memory, I/O, and communication statistics for coupled components and help explain the coupling overhead. The U. Oregon performance engineering team will provide the development team the expertise needed to efficiently observe and explain performance obtained by coupling M3D-C1, NIMROD, and XGC1 codes using the new EFFIS 2.0 framework.

**Statement of Work**

Year 1:

Install and validate TAU on OLCF Titan platform.

Instrument and measure performance of standalone XGC1 code using accelerators.

Instrument NIMROD, and M3D-C1 codes.

Create a performance database to store individual and coupled performance.

Year 2:

Design interfaces between TAU and ADIOS.

Implement ADIOS component for storing performance provenance information.

Collect performance data for coupled codes.

Year 3:

Create a performance regression test harness for coupled MHD codes using EFFIS 2.0.

Explain performance artifacts and for the coupled codes including performance variability reports.

Release performance engineered coupled MHD codes with support for collecting runtime performance statistics.